



QUARKUS

OpenAPI Api-First (Generate Sources)

JEE Microservices

@ CGS IT – 2023

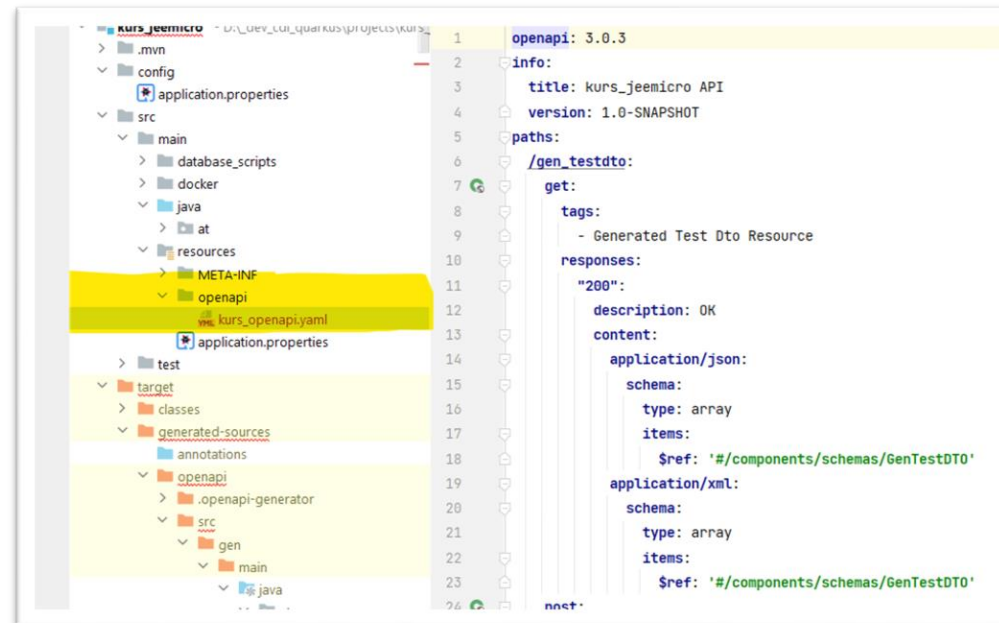
Version 1.0.5

Inhalt

- Rest Standard

OpenAPI File first

- Um aus einer OpenAPI Spezifikation den Server Java Code generieren lassen zu können, muss das API YAML File mit eingecheckt werden, um dem Generator zur Verfügung zu stehen.



Maven Konfiguration

- Mit dem openapi-generator-maven-Plugin kann aus dem OpenAPI YAML File der benötigte Source-Code generiert werden, als auch die DTOs.
- Zusätzlich wird eine Delegate Klasse generiert, die wir zur Anbindung des vollständig generierten Server Stubs an unsere Server Implementierung verwenden können.

```
<plugins>
  <plugin>
    <groupId>org.openapitools</groupId>
    <artifactId>openapi-generator-maven-plugin</artifactId>
    <version>6.4.0</version>
    <executions>
      <execution>
        <goals>
          <goal>generate</goal>
        </goals>
        <configuration>
          <inputSpec>${project.basedir}/src/main/resources/openapi/kurs_openapi.yaml</inputSpec>
          <generatorName>jaxrs-resteasy</generatorName>
          <apiPackage>at.cgsit.jeemicro.openapi.api</apiPackage>
          <modelPackage>at.cgsit.jeemicro.openapi.model</modelPackage>
          <supportingFilesToGenerate>
            ApiUtil.java
          </supportingFilesToGenerate>
          <configOptions>
            <delegatePattern>true</delegatePattern>
            <sourceFolder>src/gen/main/java</sourceFolder>
          </configOptions>
        </configuration>
      </execution>
    </executions>
  </plugin>
```

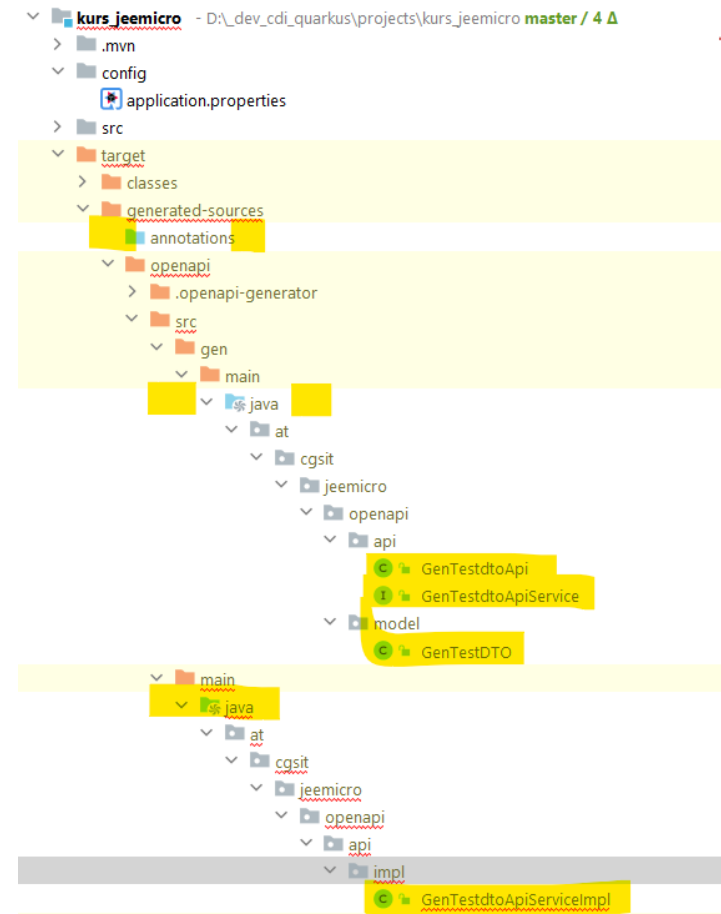
Maven Source Directory Config

- Mittels Maven Helper Plugin können die temporären target directories korrekt zum IntelliJ
- und für maven in den Classpath hinzugefügt werden.

```
<plugin>
  <groupId>org.codehaus.mojo</groupId>
  <artifactId>build-helper-maven-plugin</artifactId>
  <version>3.4.0</version>
  <executions>
    <execution>
      <id>add-source</id>
      <phase>generate-sources</phase>
      <goals>
        <goal>add-source</goal>
      </goals>
      <configuration>
        <sources>
          <source>target/generated-sources/annotations</source>
          <source>target/generated-sources/openapi/src/gen/main/java</source>
          <source>target/generated-sources/openapi/src/main/java</source>
        </sources>
      </configuration>
    </execution>
  </executions>
</plugin>
```

OpenAPI – Generierter Output

- Die Generierten OpenAPI Files inkludieren
 1. GenTestdtoAPI.java
 2. GenTestdtoAPIService.java
 3. GenTestDTO.java
 4. GenTestdtoApiServiceImpl.java
- Die Dateien sind für das IntelliJ projekt und für maven als SOURCE Folder sichtbar



Generierter Code

- Der Quellcode wird für den Server korrekt generiert.
- Mittels Generated Service Interface und Implementierung können wir das API Service bean implementieren das die Anbindung an unseren manuellen Code erledigt.

```
@Path("/gen_testdto")
@io.swagger.annotations.Api(description = "the gen_testdto API")
@javax.annotation.Generated(value =
"org.openapitools.codegen.languages.JavaResteasyServerCodegen", date =
"2023-05-30T22:30:49.429154200+02:00[Europe/Berlin]")
public class GenTestdtoApi {

    @Inject GenTestdtoApiService service;

    @GET
    @Produces({ "application/json", "application/xml" })
    @io.swagger.annotations.ApiOperation(value = "", notes = "", response =
GenTestDTO.class, responseContainer = "List", tags={ "Generated Test Dto
Resource", })
    @io.swagger.annotations.ApiResponses(value = {
        @io.swagger.annotations.ApiResponse(code = 200, message = "OK",
response = GenTestDTO.class, responseContainer = "List") })
    public Response genTestdtoGet(@Context SecurityContext securityContext)
    throws NotFoundException {
        return service.genTestdtoGet(securityContext);
    }

    @DELETE
    @Path("/{id}")
```

Limitierungen

- Mit dem Stand 30.05. kann der openapi generator die import pakete noch nicht für den neuen import pfad generieren
 - `import jakarta.ws.rs`
- Sobald dies behoben ist, kann der Generator entsprechend dem Beispiel genutzt werden.

Danke für Ihre Aufmerksamkeit